

# Quantum search algorithm tailored to clause satisfaction problems

Avatar Tulsi

Department of Physics, IIT Bombay, Mumbai-400076, India\*

Many important computer science problems can be reduced to clause satisfaction problem. We are given  $n$  Boolean variables  $x_k$  and  $m$  clauses  $c_j$  where each clause is a function of values of some of  $x_k$ s. We want to find an assignment  $i$  of  $x_k$ s for which all  $m$  clauses are satisfied. Let  $f_j(i)$  be a binary function which is 1 if  $j^{\text{th}}$  clause is satisfied by the assignment  $i$  else  $f_j(i) = 0$ . Then the solution is  $r$  for which  $f(i = r) = 1$ , where  $f(i)$  is the AND function of all  $f_j(i)$ s. In quantum computing, Grovers algorithm can be used to find  $r$ . A crucial component of this algorithm is the selective phase inversion  $I_r$  of the solution state encoding  $r$ .  $I_r$  is implemented by computing  $f(i)$  for all  $i$  in superposition which requires computing AND of all  $m$  binary functions  $f_j(i)$ s. Hence there must be coupling between the computation circuits for each  $f_j(i)$ s. In this paper, we present an alternative quantum search algorithm which relaxes the requirement of such couplings. Hence it offers implementation advantages for clause satisfaction problems.

PACS numbers: 03.67.Ac

## I. INTRODUCTION

Grovers algorithm is used to search an item satisfying certain properties out of a database of  $N$  items [1]. Let the index  $i$  ( $i \in \{0, 1, \dots, N - 1\}$ ) denote these items. Consider a quantum system of  $n = \log_2 N$  qubits with Hilbert space of dimension  $N$ , whose  $N$  basis states is used to encode  $N$  database items with one-to-one correspondence. Grovers algorithm starts with the state  $|\hat{0}\rangle$  in which all qubits are in  $|0\rangle$  state. Then it applies Walsh-Hadamard transformation  $W$  which is nothing but application of Hadamard gate on all qubits. After this, we get the state which is a uniform superposition of all basis states, i.e.

$$W|\hat{0}\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle \quad (1)$$

Grovers algorithm then successively iterates the Grover operator

$$\mathcal{G} = WI_{\hat{0}}WI_r \quad (2)$$

on above state to get the basis state  $|r\rangle$  encoding the solution of search problem.

Here  $I_{\hat{0}}$  and  $I_r$  are the selective phase inversions of these two states. Mathematically, they are written as

$$I_{\hat{0}} = \mathbb{1}_2 |\hat{0}\rangle\langle\hat{0}| ; I_r = \mathbb{1}_2 |r\rangle\langle r| . \quad (3)$$

The number of iterations of  $\mathcal{G}$  required by Grovers algorithm is  $(\pi/4)\sqrt{N}$  assuming that there is a unique solution  $r$ . This is quadratically faster than classical search algorithms which take  $O(N)$  time steps. Grovers algorithm is proved to be strictly optimal [2]. Out of the two selective transformations,  $I_{\hat{0}}$  is easy to implement as

we know the state  $|\hat{0}\rangle$ . But we dont know the solution state  $|r\rangle$  in advance and  $I_r$  is implemented using an *oracle* transformation. Basically, for all  $|i\rangle$  in superposition, the oracle computes a binary function  $f(i)$  which is 1 if  $|i\rangle = |r\rangle$  else 0. Then the computed value of  $f(i)$  is used to selectively invert the phase of  $|i = r\rangle$ .

In the clause satisfaction problems, we have  $n$  Boolean variables  $x_k$  ( $k \in \{1, 2, \dots, n\}$ ) and each variable can take two values 0 or 1. Let the index  $i$  ( $i \in \{0, 1, \dots, N - 1\}$ ) denote the different possible assignments of these variables where  $N = 2^n$ . We have  $m$  clauses  $c_j$  ( $j \in \{1, 2, \dots, m\}$ ) where satisfaction of each clause depends upon values of a subset of  $n$  variables. Typically this subset involves few number of variables. For example, in the widely studied NP-complete problem of **3SAT**, a clause is satisfied if OR of 3 terms is 1, where each term is either a Boolean variable  $x_k$  or its negation  $\bar{x}_k$  ( $\bar{x}_k = 1$  if  $x_k = 0$  else  $\bar{x}_k = 0$ ). The solution is a particular assignment  $i = r$  which satisfies all  $m$  clauses.

Obviously, in such problems, the function  $f(i)$  is AND of  $m$  different binary functions  $f_j(i)$  where  $f_j(i) = 1$  if  $j^{\text{th}}$  clause  $c_j$  is satisfied else 0. To use Grovers algorithm, we need to compute  $f(i)$  for all  $|i\rangle$  in superposition to implement  $I_r$ . As  $f(i)$  is AND of all  $f_j(i)$ s there has to be some coupling between individual computation circuits corresponding to each  $f_j(i)$ . These couplings may add significantly to physical implementation challenges of Grovers algorithm depending upon the kind of hardware that will be used for quantum computing in future.

In this paper, we present an alternative quantum search algorithm which is naturally tailored to such kind of problems. It relaxes the requirement of coupling computation circuits for each  $f_j(i)$ s as we dont need to compute AND of all  $f_j(i)$ s. In next section, we present the algorithm and we present its analysis in Section III. The analysis mainly uses the results of general quantum search algorithm presented in [3]. We then discuss and conclude in Section IV.

---

\*Electronic address: tulsi9@gmail.com

## II. ALGORITHM

Let  $\mathcal{H}_N$  denote the Hilbert space of dimension  $N = 2^n$  of  $n$  qubits where each qubit represents one of the given Boolean variables  $x_k$ . We attach an ancilla qubit to this system and let  $\mathcal{H}_2$  denote the corresponding two-dimensional Hilbert space. We work in the  $2N$  dimensional joint Hilbert space  $\mathcal{H} = \mathcal{H}_2 \otimes \mathcal{H}_N$ .

For each clause  $j$ , we define an operator  $\mathcal{D}_j$  which computes the binary function  $f_j(i)$  corresponding to  $j^{\text{th}}$  clause and then applies a controlled transformation on the ancilla qubit. If  $f_j(i) = 1$  then it leaves the ancilla qubit unchanged else it applies the single-qubit operator  $R_m$  on the ancilla qubit, where

$$R_m = \exp(i\pi/m)|0\rangle\langle 0| + \exp(-i\pi/m)|1\rangle\langle 1|. \quad (4)$$

Obviously, the operator  $\mathcal{D}_j$  acts only on  $n+1$  qubits where  $n$  is the number of Boolean variables involved in  $j^{\text{th}}$  clause. The extra one qubit is the ancilla qubit on which  $\mathcal{D}_j$  does a controlled operation. The eigenspectrum of  $\mathcal{D}_j$  is of the following form

$$\begin{aligned} \mathcal{D}_j(|0\rangle|i\rangle) &= \exp[i\pi(1-f_j(i))/m]|0\rangle|i\rangle \\ \mathcal{D}_j(|1\rangle|i\rangle) &= \exp[-i\pi(1-f_j(i))/m]|1\rangle|i\rangle. \end{aligned} \quad (5)$$

Now consider the operator

$$\mathcal{D} = \mathcal{D}_m \mathcal{D}_{m-1} \cdots \mathcal{D}_2 \mathcal{D}_1 \quad (6)$$

which is basically a product of  $\mathcal{D}_j$ s for all  $j$  from 1 to  $m$ . As each  $\mathcal{D}_j$  commute with each other, being diagonal in nature, the order of operators doesn't matter for implementing  $\mathcal{D}$ . Here we note that each unitary operator  $\mathcal{D}_j$  involves computation of  $f_j(i)$  for only one clause and for implementing  $\mathcal{D}$ , we don't need to couple any of them with each other. We just successively apply  $\mathcal{D}_j$  operators for all  $j$  in any chosen order. It is easy to see that the eigenspectrum of  $\mathcal{D}$  is given by

$$\begin{aligned} \mathcal{D}(|0\rangle|i\rangle) &= \exp[i\pi u_i/m]|0\rangle|i\rangle \\ \mathcal{D}(|1\rangle|i\rangle) &= \exp[-i\pi u_i/m]|1\rangle|i\rangle. \end{aligned} \quad (7)$$

Here  $u_i = \sum_{j=1}^m (1 - f_j(i))$  is the total number of clauses unsatisfied by the assignment  $i$ . As  $u_r = 0$  (the solution satisfies all clauses by definition), we see that  $\mathcal{D}$  has a two-dimensional degenerate eigenspace orthogonally spanned by  $|0\rangle|r\rangle$  and  $|1\rangle|r\rangle$  with eigenvalue 1.

Now we present the algorithm.

- (1) Initially, we put all qubits including the ancilla qubit in  $|0\rangle$  state.
- (2) Then we apply Walsh-Hadamard transform  $W_{n+1}$  on all  $n+1$  qubits including the ancilla qubit. We get the state  $|+\rangle_{n+1} = |+\rangle|+\rangle_n$ , where  $|+\rangle = 1/\sqrt{2}(|0\rangle + |1\rangle)$  is the ancilla qubit state and  $|+\rangle_n = (1/\sqrt{N}) \sum_i |i\rangle$  is the uniform superposition of all basis states  $|i\rangle$  of  $\mathcal{H}_N$ .
- (3) We perform  $q$  times iteration of the operator  $\mathcal{A} = W_{n+1} I_{0,\hat{0}} W_{n+1} \mathcal{D}$  on the initial state  $|+\rangle_{n+1}$ . Here  $\mathcal{D}$  is given by Eq. (7) and  $I_{0,\hat{0}}$  is the selective phase inversion

of the  $|0\rangle|\hat{0}\rangle$  state. It is easy to see that  $W_{n+1} I_{0,\hat{0}} W_{n+1}$  is nothing but the selective phase inversion  $I_{+,n+1}$  of the initial state  $|+\rangle_{n+1}$ . So we can write the operator  $\mathcal{A}$  as

$$\mathcal{A} = I_{+,n+1} \mathcal{D} \quad (8)$$

After  $q$  iterations, we get the state  $|w_q\rangle = \mathcal{A}^q |+\rangle_{n+1}$ . (4) As we show in next section, there exists a  $q$  for which  $|w_q\rangle$  is close to the state  $|+\rangle|r\rangle$  and measuring  $|w_q\rangle$  will output the solution  $r$  with a significant probability.

(5) We repeat the entire algorithm few times to get the solution with probability very close to 1.

## III. ANALYSIS

To prove the working of algorithm, we need to prove that  $\mathcal{A}^q |+\rangle_{n+1}$  is close to  $|+\rangle|r\rangle$  state for some  $q$ . To prove this, we prove the reverse operation, i.e. we prove that  $(\mathcal{A}^\dagger)^q |+\rangle|r\rangle$  is close to the initial state  $|+\rangle_{n+1}$  state. From (8), we have

$$\mathcal{A}^\dagger = \mathcal{D}^\dagger I_{+,n+1}. \quad (9)$$

Note that  $I_{+,n+1}$  is a self-inverse transformation. The above operator is just a special kind of the general quantum search operator  $\mathcal{S} = D_s I_t^\phi$  that we have analyzed in [3]. We just need to make the following substitutions in  $\mathcal{S}$ ,

$$\begin{aligned} |s\rangle &\rightarrow |+\rangle|r\rangle, & |t\rangle &\rightarrow |+\rangle_{n+1}, \\ \mathcal{S} &\rightarrow \mathcal{A}^\dagger & \phi &\rightarrow \pi. \end{aligned} \quad (10)$$

We refer the readers to original paper [3] for the details of analysis as here we just use the results of analysis.

In the analysis of [3], by convention, the initial state  $|s\rangle$  is the eigenstate of  $D_s$  with eigenvalue 1. In case of a degenerate eigenspace orthonormally spanned by the states  $|s_m\rangle$ , the initial state is chosen according to Eq. (1) of [3] as

$$|s\rangle = \frac{1}{\alpha} \sum_m \langle s_m | t \rangle |s_m\rangle, \quad \alpha^2 = \sum_m |\langle s_m | t \rangle|^2. \quad (11)$$

Note that here the notation  $m$  is used just to conform to the analysis of [3]. It should not be confused with the notation of this paper where  $m$  denotes the number of clauses. In our case,  $\mathcal{D}$  has a 2-dimensional degenerate eigenspace spanned by  $|0\rangle|r\rangle$  and  $|1\rangle|r\rangle$  with eigenvalue 1. As  $|t\rangle = |+\rangle_{n+1}$  in our case, it is easy to check that in (10), the state  $|s\rangle$  becomes  $|+\rangle|r\rangle$  just to satisfy (11). Also, it is easy to check that in our algorithm,

$$\alpha = |\langle r | W | \hat{0} \rangle| = 1/\sqrt{N}. \quad (12)$$

Using the analysis of [3] further, we find that only two eigenstates  $|\lambda_\pm\rangle$  with the corresponding eigenvalues  $e^{i\lambda_\pm}$  of  $\mathcal{A}^\dagger$  are relevant for our algorithm as the state  $|+\rangle|r\rangle$  is almost completely spanned by them. These eigenvalues

$\lambda_{\pm}$  are given by Eq. (12) of [3]. In our case, this equation becomes

$$\lambda_{\pm} = \pm \frac{2}{B\sqrt{N}} (\tan \eta)^{\pm 1} \cot 2\eta = \frac{\Lambda_1 \sqrt{N}}{2B}. \quad (13)$$

where

$$B = \sqrt{1 + \Lambda_2}, \quad \Lambda_p = \sum_{\ell \neq \psi, r} |\langle \ell | + \rangle_{n+1}|^2 \cot^p \frac{\theta_{\ell}}{2}. \quad (14)$$

Here  $|\ell\rangle$  denote the eigenstates of  $\mathcal{D}$  with corresponding eigenvalues  $\exp(i\theta_{\ell})$ . As the sum is over  $|\ell\rangle \neq |\psi\rangle$  (here  $|\psi\rangle$  is any state of ancilla qubit) and  $\theta_{\ell=\psi, r} = 0$  by convention, we need to consider only those eigenstates with non-zero eigenphases  $\theta_{\ell}$  to evaluate this sum. Using Eq. (7), we find that such eigenstates with corresponding eigenphases are

$$\begin{aligned} |0\rangle |i \neq r\rangle & , \quad \exp[i\pi u_i/m] \\ |1\rangle |i \neq r\rangle & , \quad \exp[-i\pi u_i/m]. \end{aligned} \quad (15)$$

So we have  $|\langle \ell | + \rangle_{n+1}|^2 = 1/2N$  for all  $\ell$  and we get

$$\Lambda_p = \frac{1}{2N} \sum_{\ell \neq \psi, r} \cot^p \frac{\pi u_i}{2m}. \quad (16)$$

It is easy to check that  $\Lambda_1$  vanishes because the contributions from eigenstates  $|0\rangle |i \neq r\rangle$  identically cancels that from eigenstates  $|1\rangle |i \neq r\rangle$ . Also, if  $N_u$  denotes the total number of assignments  $i$  which dont satisfy  $u$  out of  $m$  clauses, then we can write  $\Lambda_2$  as

$$\Lambda_2 = \frac{1}{N} \sum_{u=1}^m N_u \cot^2 \frac{\pi u}{2m}. \quad (17)$$

With  $\Lambda_1 = 0$ , (13) indicates that  $\eta = \frac{\pi}{4}$  and so

$$\lambda_{\pm} = \pm \frac{2}{B\sqrt{N}}. \quad (18)$$

With  $\eta = \pi/4$  and  $\phi = \pi$ , Eq. (23) and (24) of [3] gives us the initial state  $|s\rangle$  and the effect of iterating  $\mathcal{S}$  on  $|s\rangle$  in terms of two relevant eigenstates  $|\lambda_{\pm}\rangle$ . With our substitutions (??), we get

$$|+, r\rangle = -i/\sqrt{2}[e^{i\lambda_+/2}|\lambda_+\rangle - e^{i\lambda_-/2}|\lambda_-\rangle], \quad (19)$$

and

$$(\mathcal{A}^\dagger)^q |+, r\rangle = -i/\sqrt{2}[e^{iq'\lambda_+}|\lambda_+\rangle - e^{iq'\lambda_-}|\lambda_-\rangle], \quad (20)$$

where  $q' = q + \frac{1}{2}$ .

For  $q = q_m \approx \pi/2|\lambda_{\pm}| = \pi B \sqrt{N}/4$ , the state  $(\mathcal{A}^\dagger)^{q_m} |+, r\rangle$  is very close to the state given by

$$1/\sqrt{2}(|\lambda_+\rangle + |\lambda_-\rangle). \quad (21)$$

As shown in [3], the above state has an amplitude of  $1/B$  with the target state  $|t\rangle$ , which in our case is the state  $|+\rangle_{n+1}$ . So we have proved that

$$|\langle + |_{n+1} (\mathcal{A}^\dagger)^{\pi B \sqrt{N}/4} |+, r\rangle| = |\langle +, r | \mathcal{A}^{\pi B \sqrt{N}/4} |+\rangle_{n+1}| = 1/B. \quad (22)$$

Thus one running of our algorithm succeeds in finding the solution with a probability of  $1/B^2$  and  $O(B^2)$  times running of our algorithm will give the solution with a probability approaching 1. So we need a total of  $O(\pi \sqrt{N} B^3 / 4)$  iterations of  $\mathcal{A}$  to solve the search problem.

In general, for clause satisfaction problems  $B = O(1)$ . For example, in the case of 3SAT problem, we know that with probability very close to 1, a randomly picked assignment  $i$  will not satisfy  $m/8 \pm O(\sqrt{m})$  clauses. So the sum in (17) is approximately  $\cot \frac{\pi}{16} \approx 4.96 = O(1)$ .

We point out that our analysis of [3] holds only if  $|\lambda_{\pm}| \ll \theta_{\min}$ , where  $\theta_{\min}$  is the minimum eigenphase of  $\mathcal{D}$  different from 0. In our case,  $\theta_{\min} = \pi/m$  and as we found  $|\lambda_{\pm}| = O(1/\sqrt{N})$ , this assumption is satisfied as long as  $N \gg m^2$  which is true in almost all practical situations.

#### IV. DISCUSSION AND CONCLUSION

We have presented a quantum search algorithm which naturally relates to the clause satisfaction problems. This algorithm allows us to find the solution using  $O(\sqrt{N})$  oracle queries without any necessity of coupling the individual computation circuits corresponding to individual clause satisfactions. This algorithm has a potential to save the computational resources required to implement the oracle transformation  $I_r$  in Grover's search algorithm.

In spirit, our algorithm is quite similar to that presented by Kato [6]. There Kato has shown that implementation of  $I_{\hat{0}}$ , the selective phase inversion of the  $|\hat{0}\rangle$  (all qubits in  $|0\rangle$ ) state is not necessary for quantum search algorithms and this operator can be replaced by an operator made up of only single-qubit gates and so physically easier to implement. Here we have shown that implementation of the selective phase inversion of the solution state  $I_r$ , which is physically harder to implement, is not necessary for quantum search algorithms for clause satisfaction problems. Rather, it can be replaced by physically easier to implement operators.

We believe that similar ideas can be used for other kind of search problems also to design quantum search algorithms which are physically easier to implement.

---

- [1] L.K. Grover, Phys. Rev. Lett. **79**, 325 (1997).
- [2] C. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, SIAM J. Computing **26**, 1510 (1997) [arXiv.org:quant-ph/9701001].
- [3] A. Tulsi, Phys. Rev. A **86**, 042331 (2012).
- [4] L.K. Grover, Phys. Rev. Lett. **80**, 4329 (1998).
- [5] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp, Contemporary Mathematics (American Mathematical Society, Providence), **305**, 53 (2002) [arXiv.org:quant-ph/0005055].
- [6] G. Kato, Phys. Rev. A **72**, 032319 (2005).